



US006182275B1

(12) **United States Patent**
Beelitz et al.

(10) **Patent No.:** **US 6,182,275 B1**
 (45) **Date of Patent:** ***Jan. 30, 2001**

(54) **GENERATION OF A COMPATIBLE ORDER
 FOR A COMPUTER SYSTEM**

2 264 797 9/1993 (GB) G06F/15/20

(List continued on next page.)

(75) **Inventors:** Alan E. Beelitz, Austin; Paul J. Mala,
 Cedar Park, both of TX (US)

OTHER PUBLICATIONS

Gateway 2000, Inc. (Hardest site to work through), Com-
 puterworld, v31, n46, p. 108, Nov. 1997.*

(List continued on next page.)

(73) **Assignee:** Dell USA, L.P., Round Rock, TX (US)

(*) **Notice:** This patent issued on a continued pro-
 secution application filed under 37 CFR
 1.53(d), and is subject to the twenty year
 patent term provisions of 35 U.S.C.
 154(a)(2).

Primary Examiner—Tariq R. Hafiz

Assistant Examiner—Hoang-Vu Antony Nguyen-Ba

(74) *Attorney, Agent, or Firm*—Haynes and Boone, L.L.P.

(57) **ABSTRACT**

Under 35 U.S.C. 154(b), the term of this
 patent shall be extended for 0 days.

A system for specifying, ordering, and building a build-to-
 order computer system. After initiating an ordering session,
 a user such as a purchaser or designer is presented with a list
 of options such as a list of operating systems offered by a
 computer system vendor or manufacturer that may be imple-
 mented on a targeted computer system. After receiving an
 indication of a selection from a first list of options, the
 system accesses a computer system readable master data
 base to generate a second list of options such as software
 programs wherein each option of the second list is compat-
 ible with the selection from the first list. The master data
 base includes entries for every option offered by the com-
 puter system vendor or manufacturer and includes at least
 one tag indicating compatibly with other entries in the
 master data base. The system can be used to present to the
 user a plurality of lists wherein all of the options presented
 are compatible with the previous selections. The system
 writes indications of the selections in a data file. The data file
 is provided to manufacturing wherein the selections are
 implemented on a targeted computer system using the data
 file. The system may also include a sniffing feature used to
 determine particular hardware parameters of the targeted
 computer system. The system uses the determined param-
 eters in generating the compatible lists of options. The
 system enables a purchaser to buy and order a computer
 system over a computer network such as the Internet.

(21) **Appl. No.:** 09/012,962

(22) **Filed:** Jan. 26, 1998

(51) **Int. Cl.⁷** G06F 9/45

(52) **U.S. Cl.** 717/1; 707/4; 707/10;
 707/104

(58) **Field of Search** 395/712; 705/26;
 717/11, 1; 707/4, 10, 104

(56) **References Cited**

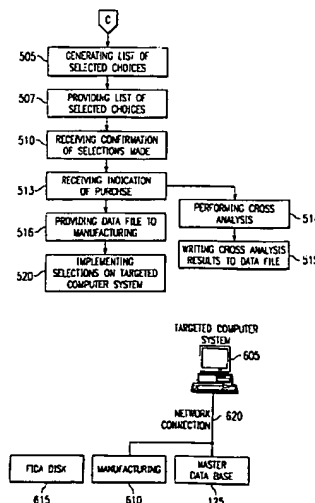
U.S. PATENT DOCUMENTS

4,984,155 * 1/1991 Geier et al. 705/26
 5,003,472 * 3/1991 Perrill et al. 705/15
 5,544,040 * 8/1996 Gerbaulet 705/26
 5,570,292 * 10/1996 Abraham et al. 364/473.01
 5,665,951 * 9/1997 Newman et al. 235/375
 5,808,894 * 9/1998 Wiens et al. 364/479.01

FOREIGN PATENT DOCUMENTS

2 234 097 1/1991 (GB) G06F/15/20

23 Claims, 6 Drawing Sheets



DOCUMENT-IDENTIFIER: US 6182275 B1

TITLE: Generation of a compatible order for a computer system

----- KWIC -----

ABPL:

A system for specifying, ordering, and building a build-to-order computer system. After initiating an ordering session, a user such as a purchaser or designer is presented with a list of options such as a list of operating systems offered by a computer system vendor or manufacturer that may be implemented on a targeted computer system. After receiving an indication of a selection from a first list of options, the system accesses a computer system readable master data base to generate a second list of options such as software programs wherein each option of the second list is compatible with the selection from the first list. The master data base includes entries for every option offered by the computer system vendor or manufacturer and includes at least one tag indicating compatibility with other entries in the master data base. The system can be used to present to the user a plurality of lists wherein all of the options presented are compatible with the previous selections. The system writes indications of the selections in a data file. ~~The data file is provided to manufacturing wherein the selections are implemented on a targeted computer system using the data file.~~ The system may also include a sniffing feature used to determine particular hardware parameters of the targeted computer system. The system uses the determined parameters in generating the compatible lists of options. The system enables a purchaser to buy and order a computer system over a computer network such as the Internet.

BSPR:

By presenting the user via a user interface with only compatible choices, the user does not have to be sophisticated in the language of computer equipment or computer software programs to be able to specify and/or order a build-to-order computer system. Thus, a computer system vendor incorporating the present invention does not have to employ highly trained sales staff. Also, such a system allows a computer system vendor to incorporate an automated system to sell build-to-order computer systems over a communications network, such as a push key telephone system, or over the Internet.

DEPR:

FIG. 1 is a block diagram of one embodiment of a system for generating a compatible order for a computer system and for building a computer system according to the compatible order. The system includes a control computer system 103 having an associated control memory 107. Control computer system 103 (control) executes a computer program or programs designed to generate a computer readable data file that is used to manufacture a targeted computer system according to the specifications of a purchaser or designer of the targeted computer system. In the embodiment shown, a user such as a purchaser or designer places an order to buy a computer system via a user interface which in the embodiment shown, is a computer terminal 105. Terminal 105 accesses the control computer system 103 via a computer network 110, which in the embodiment shown, is a wide area network (WAN) such as the Internet or intranet. In one embodiment, terminal 105 executes an internet browser program. In other embodiments, terminal 105 may be coupled to control computer system 103 via a local area network (LAN). In other embodiments, the terminal is operated by a representative (employee or agent) of the computer system vendor who receives instructions from the consumer either in person or over the telephone. In other embodiments, the control computer system is a personal computer system and the user interface is the keyboard, display or monitor, and mouse of the personal computer system. In other embodiments, a user such as a purchaser may enter their choices via a user interface such as a telephone that is operably coupled to control via a telephone network.

DEPR:

After receiving an indication via the network connection 110 from terminal 105 that a customer desires to purchase a computer system, control 103 accesses a master data base 125 via a network connection 120 to generate a list of options available for a build-to-order computer system as offered by the manufacturer or computer system vendor. The control computer system 103 then provides the

DEPR:

The tags in the master data base can also be used to indicate to control that other operations need to be performed by control based upon a previous selected choice by the user. Also, a tag or tag field may be used to indicate that the computer system vendor offers a variety of options for a particular entry (either a software program, or hardware component, or implementation operation). Consequently, when the control computer system reads the tag, it then knows to create a list of options for the particular entry if that entry is selected. For example, the entry for an operating system may include a tag that indicates that the vendor offers a standard disk format or an alternative disk format with the operating system. To create this list, the control computer system scans the master data base to look for other tags of a particular type to construct a list of disk format alternatives. Consequently, the tag indicates to the control computer system that other options need to be presented to the user via the user interface and that the user needs to select an option. In other embodiments, the tags would indicate what hardware components, software programs, installation instructions or operations, tests, and other implementation operations are required for the implementation of the entry onto a computer system. In one embodiment, the particular tag would indicate to control to go to another data base to perform a particular function such as loading text onto the data file from another data base. Tags may also be used to indicate that an entry is incompatible with other entries.

DEPR:

Although in FIG. 1, only one control computer system 103 is shown accessing the master data base 125, in other embodiments, multiple control computer systems can access the master data base 125 to conduct ordering or specifying sessions with a plurality of users. In one embodiment, the control computer systems are personal computers of the inside sales staff of a computer system vendor. One advantage of having one central master data base is that the lists generated by the control computer systems are compatible and up-to-date. Furthermore, having a master data base allows for the easy update of the system in offering new computer hardware components and software program applications. For a vendor to offer a new software program or hardware component, a new entry, including tags to indicate compatibility with other entries, is created on the master data base. The specifying programs run by the control computer systems do not have to be updated in order to offer a new component, operation, or program.

DEPR:

The system of FIG. 1 is a convenient system for allowing a purchaser to order a build-to-order computer system and for a manufacturer to build the computer system as specified.

DEPR:

FIGS. 2-5 are flow diagrams showing steps for generating a compatible order for a computer system and for building the computer system according to that order. These flow diagrams are executed by the control computer system 103 and, in one embodiment, are stored in the control memory 107. In one embodiment, the flow diagrams are implemented by a computer program written in the C programming language. In step 201, the control computer system 103 receives an indication from a user interface 105 indicating the desire to purchase a computer system. This occurs, for example, when a purchaser accesses the computer system vendor's web site via the Internet and activates a button to initiate a specifying or ordering session.

DEPR:

In another embodiment, the entries for all of the operating systems listed in the master data base include a version tag. In this embodiment, the second list presented to the user would include all of the versions offered for the selected operating system type. Those versions on the second list are determined by the version tags for the entries having the selected operating system type tag. Consequently in this embodiment, a particular version would appear on the second list if there is at least one entry that has both the tag of the selected operating system type and the tag of the particular version. After receiving from the user interface an indication of which version is selected, control would access the master data base to create a third list (step not shown). This third list would include all operating systems that have the tag of the selected operating system type and of the tag of the selected version. This third list would represent all of the languages offered

by the computer system vendor for the selected version of the selected operating system type.

DEPR:

In step 238, control 103 determines if there are any "always run parts" associated with the selected version and language of the selected operating system type. Associated always run parts for a selected choice are software programs, installation operations, testing operations, or other implementation operations that are required for or that the manufacture always performs or implements with the implementation of the selected choice on a targeted computer system. Examples of associated always run parts for the selected version and language of the selected operating system type include required patches, required hard drive preparation operations, and tests to be performed on the targeted computer system before and/or after the selected operating system has been installed.

DEPR:

In step 310, control generates a list of patches available for the selected version and language of the selected operating system type. Patches are software programs that are available for a selected choice that correct a problem with a specific choice or enhance the performance of the selected choice. For example, if an operating system provider provides the operating system with the Direct Memory Access (DMA) deactivated, the computer system vendor may offer a patch to activate the DMA to improve performance. In other embodiments, patches may be used to implement the selected version and language of the selected operating system type with a particular hardware component that was selected by the user in a previous choice. However, if a patch is required, then the patch would be considered an always run part and an indication of the patch would be written to the data file in step 241 of FIG. 2.

DEPR:

In some embodiments, control generates a list of always run parts (either by accessing entries in the master data base 125 or accessing a dependency file) associated with the selected the hard drive preparations operations and writes an indication of the list of always run parts to the data file.

DEPR:

In step 318, control generates a list of always run parts associated with the selected patches. In step 320 control writes that list to the data file. Control generates the list of always run parts by accessing entries in the master data base 125 by or accessing a dependency file. In other embodiments, control 103 may complete steps 318 and 320 after the completion of the specifying session with the user. In other embodiments, steps 318 and 320 may be completed by other computer systems in manufacturing.

DEPR:

In some embodiments, the lists of software programs provided to the user interface are broken up into different sub-lists or sub-menus. For example, a first sub-list may include a list of compatible word processing programs such as MICROSOFT WORD.TM. or WORDPERFECT.TM.. Another sub-list would include computer games offered by the computer system vendor. It is understood that a variety of sub-menus classifying different software programs may be presented. For instance, internet access programs, spreadsheets, and drawing programs may be presented in different sub-menus. Breaking the list of available programs into sub-menus aids the user in selecting software programs. In one embodiment, each entry for a software program would include a tag field or tag indicating the type of software program. Control would read these tags and generate the sub-menu lists according to the software type tags. In another embodiment, each different group of software programs would be stored in a different logical section of the data base. Control would generate each sub-menu list by reading only one logical section of the data base at a time. In this embodiment, every entry in a logical section of the master data base having a tag indicating compatibility with the selected operating system would be presented in the sub menu list.

DEPR:

In step 417, control generates a list of always run parts associated with the selected software programs. In step 420 control writes that list to the data file. Control generates the list of always run parts by accessing entries in the master data base 125 by or accessing a dependency file. In other embodiments, control 103 may complete steps 417 and 420 after the completion of

list via the network connection 110 to the terminal 105 where, in the embodiment shown, the list is displayed on the terminal screen. After the user makes a selection from the list, an indication of that selected choice is sent back to the control computer system 103. The control computer system 103 then accesses the master data base 125 to generate a second list of options wherein each option of the second list is compatible with the previous selection. The control computer system then provides the second list to terminal 105 via the network connection 110 to enable the user to select from a list of options that are compatible with the previous selection.

DEPR:

Presenting a user with a list of options that are compatible with a previous selected choice advantageously allows a user to select from only those items that are compatible with the previous choice. This advantageously eases the convenience of ordering or specifying a computer system on the user in that the user does not have to remake a selection in the event that a previous choice is incompatible. It also saves the user time in that a user does not have to agonize over options that are not compatible with a previous choice. Furthermore, the orders generated are compatible as of the completion of the ordering or specifying session with the purchaser. Thus, a computer system vendor does not have to reinitiate a specifying session if the initial order is later determined to be incompatible.

DEPR:

The control computer system 103 provides the data file via a network connection 130 to manufacturing 135. Manufacturing 135 is typically located at the plant or manufacturing facility where the computer systems are assembled. The data file is used to manufacture a targeted computer system as per the selections made by the user as indicated in the data file. In one embodiment, the data file generated by the control computer system 103 is used to generate other files used in manufacturing targeted computer systems. For example, see "SOFTWARE INSTALLATION AND TESTING FOR A BUILD-TO-ORDER COMPUTER SYSTEM", Ser. No. 08/919,959, filed Aug. 29, 1997, with Richard D. Amberg, Roger W. Wong, and Michael A. Brunbridge named as inventors, which is hereby incorporated by reference in its entirety. In another embodiment, the data file is loaded directly onto a data server in manufacturing 135. When the targeted computer system 137 first boots up, it loads software and performs other operations and instructions as per the data file.

DEPR:

In one embodiment, manufacturing 135 includes a computer program called a "parser" that reads the indications or entries in the data file and associates executable shell script files located in manufacturing 135 with the indications or entries in the data file. The parser may also execute the shell script programs to execute the installation of the operating system software application programs, or other application programs on the targeted computer system 137. In one embodiment, the shell script files are stored on the master data base wherein the computer system executing the parser program has access to the master data base 125.

DEPR:

In the embodiment shown, master data base 125 is located on a server that includes a plurality of drives. In one embodiment, the master data base 125 is a computer system readable data base that includes an entry for every software program and hardware component option offered by a computer system manufacturer or vendor for a build-to-order computer system. In some embodiments, the master data base 125 also contains an entry for each implementation operation such as installation programs, hard disk drive formatting operations, and testing operations for implementing the hardware components and software programs on a computer system. These entries may be located in different files in the master data base 125. In one embodiment, the master data base is realized as a flat file.

DEPR:

Each entry in the master data base 125 includes a manufacturer specific part number or identification field with each individual hardware components, software programs, or installation operations having a specific number. Each entry may also include a description field that indicates text to be displayed on the user interface of terminal 105 when the entry is presented to the user. FIG. 8A is an alpha-numeric representation of one embodiment of entries in a parts file located in the master data base.

the specifying session with the user. In other embodiments, steps 417 and 420 may be completed by other computer systems in manufacturing.

DEPR:

In step 436, control generates a list of always run parts associated with the selected software programs from the second list. In step 439, control writes that list of always run parts to the data file. Control generates the list of always run parts by accessing entries in the master data base 125 or by accessing a dependency file. In other embodiments, control 103 may complete steps 436 and 439 after the completion of the specifying session with the user. In other embodiments, steps 436 and 439 may be completed by other computer systems in manufacturing.

DEPR:

Referring to FIG. 5, after the software program selections are made in FIG. 4, control transfers to step 505, where it accesses the data file to generate a list of all of the previously selected choices as indicated by the data file. This list reflects all of the selections made by the purchaser or user and reflects the type of computer system that will be built by the manufacturer. In step 507, control provides the list to the user interface 107 which presents the list to the user, where the user can then approve or disapprove of the choices made. If the user does not agree with a selection, the user can send an indication to control to "move backwards" in the program to the selection that the user objects to and ask for a re-selection of the part. Upon receiving an indication for the re-selection of a particular part, control re-presents the list that included the selected part either by locating in control memory 107 the saved list or by reading the previous choices from the data file to return to the preceding step that generated the list. All subsequent selections from the objected choice would be voided. Control would proceed from the point in the program where the new choice is made to ensure that all subsequent choices are compatible with the new choice.

DEPR:

In some embodiments, control generates and writes to the SDR file a list of always run parts that are implemented in every computer system that a computer system vendor makes. In one embodiment, the entry for each of these always run parts includes a tag indicating that the part is to be automatically implemented in a computer system.

DEPR:

In the embodiments shown in FIGS. 2-5, the data file generated by control is an SDR file. ~~An SDR file is a computer readable text file that includes an entry or line for each hardware component, software program, patch, or other operation to be implemented on the targeted computer system 137.~~ Each entry of the SDR includes the manufacturer's specific part number and associated prefix for the item represented. In some embodiments, the associated prefixes identify the type of part, program, or component that the entry represents. In some embodiments, each entry may include an identifying tag. An SDR file is formatted to be transferred to and read by another computer system in manufacturing 135 that executes a parser computer program to build the computer system as per the data file. The SDR file would also include customer specific information such as the name and address of the computer purchaser.

DEPR:

In another embodiment, control inputs a BAR file containing specifications for a specific computer system and enters those specifications into the SDR data file during the specifying session. For example, control may have access to several BAR files (located in another portion of the master data base 125 or in other data base) wherein each BAR file contains the hardware default specifications for a particular model of computer system offered by the computer system manufacturer. The user may be initially asked to choose a model from a list of models available. After receiving an indication of the model chosen from the user interface 105, control accesses the corresponding BAR file for that model and converts those specifications into the SDR file. Afterwards, control executes the specifying program of FIGS. 2-4 wherein the user specifies the operating system and software programs. However, because the SDR contains entries for the hardware components specified in the BAR file, the choices presented to the user in FIGS. 2-4 are also compatible with the hardware components as specified in the BAR file. In some embodiments, control presents the components as specified in the BAR file as default selections and allows the user to make changes to the default selections.

DEPR:

In step 516, control provides the data file to manufacturing 135 which in step 520 uses the data file to implement the selections as per the data file. In one embodiment, another computer system in manufacturing (internal to manufacturing 135) executes a parser software program that associates an executable shell script file with each entry of the SDR and executes the shell script file in implementing the entry on the targeted computer system 137. Thus, the selected operating system, selected software programs, selected patches, selected hard disk drive operations, the determined always run parts, and other selections are installed, implemented and tested by the execution of the associated shell script files. The parser program scans through the system descriptor record line by line and associates a shell script file with each part number. The parser program executes the shell script files by supplying or turning over the shell script files to a script processing program. The script processing program interprets the instructions of the shell script files and acts upon those instructions to perform the actual instructions for the installation or implementation of the part on the computer system. In some embodiments, the parser program may also generate and attach a suffix to a particular part number. The parser program generates the suffix by reading the identifying tag of the SDR entry or by accessing tables with the part number. FIG. 8C is an alpha-numeric representation of a computer system readable file containing a list of shell scripts files.

DEPR:

In one embodiment, the targeted computer system is connected through a network card to a server in manufacturing. The targeted computer system 137 is initially booted up to perform the operations and instructions as per associated shell script files to load the selected programs onto its hard drive and to run the tests. In one embodiment, the selected software programs and operating system can be down loaded and installed on the targeted computer system via the Internet.

DEPR:

In other embodiments, the data file may be used to create a floppy disk, which would be used to boot up the targeted computer system in the manufacturing process and install the selected programs and components on the targeted computer system.

DEPR:

In other embodiments, manufacturing 135 would use the data file to write to a floppy disk or CD ROM a duplicate set of selected software programs. This disk would also include the installation instructions and operations to load the programs from the CD ROM to the hard drive. Such a feature would advantageously provide a back up to the hard drive of the targeted computer system and, in the event of a hard drive crash, provide a convenient system to reprogramming the hard drive according to the selections specified.

DEPR:

FIG. 6 is a flow diagram showing steps that enable a user to specify both compatible hardware components and software programs for the targeted system. In step 601, control 103 receives an indication from a user interface that a purchaser wishes to buy a build-to-order computer system. In step 604, control accesses the master data base 125 to generate a list of central processor units (CPU) offered by the computer system manufacturer in a targeted computer system. This list is generated by the control computer system 103 reading the tags in the master data base 125. Each CPU offered by the computer system vendor has an entry in the master data base that includes a manufacture specific part number and associated tags. In the embodiment of FIG. 6, the tags for each processor would include an initiation tag. Every CPU whose entry includes an initiation tag would be presented in the first list. In other embodiments, each processor entry includes a type tag where the first list presented to the user would be the types of CPUs offered by the computer system vendor.

DEPR:

After the user has selected a CPU, the user interface 105 provides an indication of the selected CPU to control. In step 613, control accesses the master data base 125 to generate a list of RAM sizes and types that are available and compatible with the selected CPU. The entries for each RAM include at least one tag indicating compatibility with a CPU. The list

generated would include each RAM whose entries include a tag indicating compatibility with the selected CPU. In other embodiments, the entries for the RAMs would include other tags such as a size tag. This size tag is used to present to the user a list of compatible RAM sizes offered by the computer system vendor.

DEPR:

In FIG. 7, the targeted computer system is booted up with an operating system program on disk 615 which in the embodiment shown is a Factory Installed DOS Application (FIDA) program on a floppy disk. This FIDA program enables the target computer system 605 to invoke a software selection program (similar to that shown in FIGS. 2-5) that is initially stored in manufacturing 610. During this software selection program, the targeted computer system 605 is also the control computer system. The targeted computer system 605 accesses via network connection 620 the master data base as per the software selection program to present lists of software programs that are compatible with the previous selections made by the operator of the targeted computer system 605. In this embodiment, the lists would be displayed on the display screen or other type of user interface of the targeted computer system 605 and the user would indicate their choices via a keyboard, mouse, or other type of user interface input device.

DEPR:

After completing the software selection ordering session and generating a data file such as a system descriptor file, the data file is provided to manufacturing 610. In one embodiment, manufacturing includes other computer programs such as a parser. The targeted computer system, executing the parser program, associates the appropriate shell script with each entry in the SDR and then invokes the shell script program to implement and execute the selected programs and/or operations on the targeted computer system 605. The parser program and executable shell script files are initially stored in a data base in manufacturing 620. In one embodiment, another computer system may run the parser program to load software programs onto a hard disk drive. The hard disk drive is then sent to a customer for use in the targeted computer system.

CLPR:

3. The method of claim 1, wherein each of the third plurality of options represents an option offered by a computer system vendor of the selected choice.

CLPV:

associating an executable file with the part number of the selected choice, the executable file for use during manufacture of the computer system.

CLPV:

installing the software program in a computer system during manufacturing of the computer system according to the data file.

CLPV:

indicating in the computer readable data file at least one implementation operation required for the implementation of the selected choice in the computer system during manufacturing of the computer system.

CLPV:

wherein the means for implementing includes means for associating an executable file with the indication of the at least one selected choice and includes means for executing the executable file during manufacturing to implement the at least one selected choice on the computer system.

CLPW:

executing the first executable file during manufacturing to implement in the computer system the selected choice from the first list; and

CLPW:

executing the second executable file during manufacturing to implement in the computer system the at least one selected choice from the second list.

CCXR:

707/10

CCXR: